

**TECHNICAL
RESEARCH
REPORT**

*Institute for
Systems
Research*

**On the Tuning of Nonlinear Model
Predictive Control Algorithms**

by E. Ali and E. Zafiriou

*The Institute for Systems
Research is supported by the
National Science Foundation
Engineering Research Center
Program (NSFD CD 8803012),
Industry and the University*

TR 93-26

On the Tuning of Nonlinear Model Predictive Control Algorithms

Emad Ali and Evangelos Zafiriou*

Department of Chemical Engineering and Institute for Systems Research
University of Maryland, College Park, MD 20742

Abstract

Nonlinear Model Predictive Controllers determine appropriate control actions by solving an on-line optimization problem. A nonlinear process model is utilized for on-line prediction, making such algorithms particularly appropriate for the control of chemical reactors. The algorithm presented in this paper incorporates an Extended Kalman Filter, which allows operations around unstable steady-state points. The paper proposes a formalization of the procedure for tuning the several parameters of the control algorithm. This is accomplished by specifying time-domain performance criteria and using an interactive multi-objective optimization package off-line to determine parameter values that satisfy these criteria. A reactor example is used to demonstrate the effectiveness of the proposed on-line algorithm and off-line tuning procedure.

1. Introduction

In order to meet the increasing needs of designing control systems that take into account the nonlinear process characteristics, a number of Model Predictive Control (MPC) algorithms have emerged in the last decade, which directly utilize nonlinear models for on-line prediction. A description of various MPC algorithms is given in a review paper by Bequette [2].

The performance may deteriorate in the presence of model-plant mismatch. One attempt to address this issue is to couple the NLMPC algorithm with an optimization-based parameter estimation method as reported by Wright *et al.* [19], Li and Biegler [12], and Eaton and Rawlings [5]. Another approach is the combined parameter and state estimation via nonlinear programming. This algorithm is studied by Jang *et al.* [10] and Sistu and Bequette [3]. An alternative way to compensate for the impact of model uncertainty is by augmenting the controller with a state estimator by Kalman Filtering as proposed by Ricker [16] for linear MPC. This approach was successfully extended by Gattu and Zafiriou [7] to Nonlinear Quadratic Dynamic Matrix Control [6] for disturbance rejection of open-loop unstable processes. In this study, we show that it is advantageous to couple our NLMPC algorithm with on-line Extended Kalman Filter (EKF).

Li *et al.* [13], and Gattu and Zafiriou [8] used the concept of contraction mapping to establish sufficient closed-loop stability conditions for their algorithms for the case of open-loop stable plants. These conditions, however, are not useful from a practical point of view, since they are usually conservative [8]. The situation is much more complicated when constraints are included in the on-line optimization. For linear process dynamics, Zafiriou and Marchal [20] proved the presence of hard constraints in the MPC algorithm can lead to instability, even though the unconstrained algorithm may be stable.

The difficulties in developing theoretical conditions that guarantee stability and good performance of NLMPC, especially in the presence of disturbances, model uncertainty, and output constraints, lead designers to trial-and-error tuning of the NLMPC parameters. The trade-off problem, however, for several competing objectives can be quite complex.

This paper attempts a formalization of what is currently a trial-and-error procedure for NLMPC parameter tuning. The real-valued parameters of the algorithm are determined by an off-line optimization. While the integer parameters such as the prediction horizon and the control horizon are found by grid search. The objective of the off-line optimization is to ensure that certain performance specifications, e.g., required speed of response and limited overshoot, are satisfied in the presence of modeling error and disturbances that lie within certain maximum bounds. The off-line problem is solved by an interactive multi-objective optimization tool called CONSOLE, developed by Tits *et al.* [18].

2. NLMPC Algorithm

On-line Optimization

The algorithm finds a sequence of M future manipulated variables by minimizing on-line an objective function based on the desired output trajectories over a prediction horizon P . After the optimization, the first element of this future sequence is implemented. Then at the next sampling time, after a new measurement has been obtained, a new optimization is carried out. The objective function is as follows:

*Author to whom correspondence should be addressed. E-mail: zafiriou@src.umd.edu

$$\min_{u(t_k), \dots, u(t_{k+M-1})} \sum_{l=1}^P \|\Gamma e(t_{k+l})\|^2 + \sum_{l=1}^M \|\Delta u(t_{k+l-1})\|^2 \quad (1)$$

where k denotes the current sampling point, $t_i = iT$, with T the sampling time, and $\|\cdot\|$ the Euclidean vector norm. The predicted error is defined as $e(t_{k+l}) = y(t_{k+l}) - r(t_{k+l}) + d(t_{k+l})$, where r is the setpoint, y is the model output, and d is the deviation of the process measurement from the model output. Since future measurements are not known, the disturbance $d(t_{k+l})$, for $l = 1, \dots, P$, is considered constant in the future and equal to $d(t_k)$. The inputs u are constant between sampling points. Δu indicates the change in manipulated inputs ($\Delta u(t_{k+l}) = u(t_{k+l}) - u(t_{k+l-1})$). The inputs are assumed constant after $k + M - 1$, i.e., $\Delta u(t_{k+i}) = 0$, $i > M$. Γ and D are the diagonal matrices of weights on the outputs and the change of manipulated variables respectively. Constraints on both u and Δu are also included in the optimization problem. The optimization is carried out with the NPSOL software, written by Gill *et al.* [9], which uses a Successive Quadratic Programming algorithm.

The output prediction is obtained via numerical integration of the nonlinear model differential equations for specified inputs, using the software package DASSL [15]. Incorporation of state estimation in the output prediction to compensate for model-plant mismatch and disturbances is discussed in the following section.

State Estimation

State resetting is achieved with an Extended Kalman Filter (EKF) which allows an *additive* state correction formulation. The correction term is then used to reset not only the current state variables, but also the state variables at each future sampling point during the prediction. The construction of the EKF for nonlinear models is discussed in Lewis [11]. Here we extend the calculational procedure of the filter gain [11] to evaluate its steady state value. The correction term is then equal to the product of the calculated steady state filter gain and the deviation of the current predicted output from its actual measured value.

Let the model equations be represented by:

$$\dot{x} = f(x, u, t) + w \quad (2)$$

$$y = h(x) + v \quad (3)$$

where $w \sim (0, Q)$ and $v \sim (0, R)$ are white Gaussian noise processes assumed to be independent of each other, and to characterize the unmeasured disturbances and the measurement noise respectively. Q and R are the respective covariances of w and v and they are assumed to be diagonal matrices of the form $Q = q^2 I$ and $R = r^2 I$. In the absence of accurate knowledge of the disturbance and noise characteristics, we further simplify the filter tuning method. Defining $\sigma^2 = q^2/r^2$ and letting $r^2 = 1.0$ will uniquely

determine the Kalman Filter gain and simplify its tuning to determining only one parameter [17].

The model output prediction is described by the following steps:

step 1: Initialization. Known at the current sampling time, k , are the plant measurement $\tilde{y}(t_k)$, the model state vector $x(t_k)$, and the manipulated variable vector $u(t_{k-1})$. Set $P_k = I$.

step 2: Linearization. Obtain the following jacobians:

$$A_k = \nabla_x f(x, u, t) \big|_{x=x(t_k), u=u(t_{k-1}), t=t_k} \quad (4)$$

$$H_k = \nabla_x h(x) \big|_{x=x(t_k)} \quad (5)$$

step 3: Time update. Integrate numerically the error covariance equation:

$$\dot{P}(t) = A_k P(t) + P(t) A_k^T + Q \quad (6)$$

for one sampling interval using DASSL, with initial condition P_k . Define the solution as \bar{P}_k .

step 4: Measurement update. Compute the Kalman Gain:

$$K_k = \bar{P}_k H_k^T [H_k \bar{P}_k H_k^T + R]^{-1} \quad (7)$$

and update the error covariance:

$$P_k = [I - K_k H_k] \bar{P}_k \quad (8)$$

Repeat steps 3 and 4 till steady state value of P_k is reached.

step 5: Correction factor. Compute the Kalman Filter correction factor :

$$F_k = K_k [\tilde{y}_k - h(x(t_k))] \quad (9)$$

step 6: Output prediction. Set $\hat{x}(t_k) = x(t_k) + F_k$ and integrate the state equation over one sampling time to get $x(t_{k+1})$, then reset the value of the new state to $\hat{x}(t_k)$ by adding F_k , and evaluate the output $y(t_{k+1}) = h(\hat{x}(t_{k+1}))$. Repeat the last step over the prediction horizon P .

3. Tuning Procedure

This section formulates the tuning parameter selection problem as an off-line optimization carried out with the interactive CONSOLE software. This is a flexible formulation that allows one to use several types of performance criteria. The off-line optimization can, e.g., determine the tuning parameter values that make the closed-loop response stay within a preset constraint envelope over defined time domain as shown in Fig. 1. The envelope is used to represent various performance objectives. For example, constraints can be used to limit overshoot or undershoot, and/or maintain desired response speed. The values of constraints can be specified as functions of the setpoint change values, e.g., proportional to them. Alternatively one could choose to directly minimize the maximum possible overshoot [21].

The procedure can be described by the loop shown in Fig. 2. The desired performance specifications, e.g., the

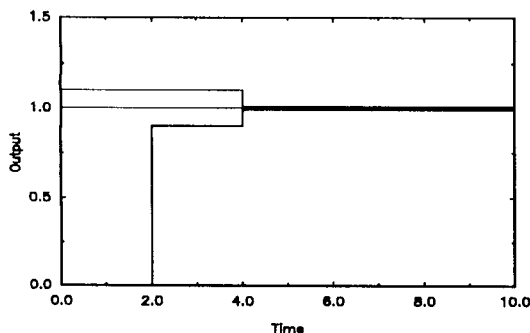


Figure 1: Desired response profile for setpoint tracking

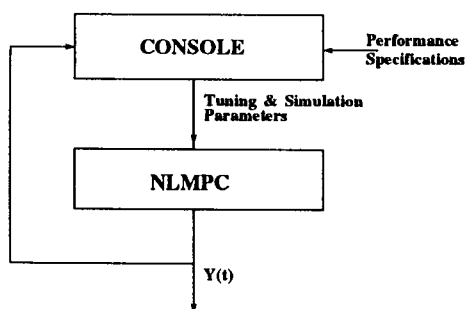


Figure 2: Tuning procedure

envelope of Fig. 1 are given to CONSOLE. Also possible setpoint changes, disturbances, model parameter errors etc, can be specified. By discretizing the range of possible values of these quantities one can define multiple objectives or constraints for CONSOLE, each of which corresponds to a particular set of values, and require that the specified performance criteria be satisfied or optimized. CONSOLE will then find suitable tuning parameter values that force every NLMPC response obtained for different discretized value of modeling error, setpoints, or disturbances to lie within the maximum bounds.

During the optimization carried out by CONSOLE, simulations of the closed-loop control system under NLMPC have to be run repeatedly for different tuning parameters, setpoints, disturbances, as well as model and plant. CONSOLE determines the next set of values to be tried, based on optimization theory, instead of trial-and-error. An added feature of CONSOLE is its interactive nature, which allows the designer to specify “good” and “bad” values for each performance constraint specification and interactively change them if CONSOLE can not find tuning parameter values to satisfy them. More detailed discussion is given in Nye and Tits [14].

It should be emphasized that CONSOLE treats all design variables as continuous real variables. Since the NLMPC algorithm uses only integer values for M and P , the optimal M and P are determined, in this paper, by performing a grid search. For each grid point (i.e., each fixed values of M and P) CONSOLE is used to determine the real-valued parameters. Note, also, that beyond the usual MPC tuning parameter, the sampling time (T), and the covariance ratio

(σ) can also be used as such parameters.

4. Illustration

A catalytic CSTR example is studied. In all simulations, the effect of the upper and lower performance constraints are balanced by equating the difference between their “good” and “bad” values. In this case CONSOLE will try to satisfy both bounds equally.

For each off-line optimization solved with CONSOLE we report the total number of NLMPC simulations. Several NLMPC simulations (corresponding to different values of disturbances) are required every time that a new point in the variable space (NLMPC tuning parameter space) has to be tried. This includes simulations made for numerical derivative computations. CONSOLE uses a forward difference formula for each parameter. The number of total NLMPC simulations is provided as an alternative to CPU time. The type of software that one uses for solving the on-line NLP has a significant effect on CPU time for each simulation.

The example is taken from the paper by Brengel and Seider [4]. An exothermic catalytic reaction in the form of $A + B \rightarrow P$ is taking place. The reactor model is:

$$\dot{x}_1 = u_1 + u_2 - 0.2x_1^{0.5} \quad (10)$$

$$\dot{x}_2 = \frac{(C_{b1} - x_2)u_1}{x_1} + \frac{(C_{b2} - x_2)u_2}{x_1} - \frac{k_1 x_2}{(1 + k_2 x_2)^2} \quad (11)$$

The process outputs are $y_1 = x_1$ (tank level), and $y_2 = x_2$ (concentration of B in the reactor). C_{b1}, C_{b2} are the concentrations in the inlet feeds of condensed and dilute B . The corresponding flow rates u_1 and u_2 are the manipulated variables. The model parameter values are $k_1 = k_2 = 1$, $C_{b1} = 24.9$, and $C_{b2} = 0.1$.

The control goal is to move the process from initial stable steady state conditions of $u_1 = u_2 = 1$, $y_1 = 40$, and $y_2 = 0.4$, to a new unstable steady state at $y_1 = 100$ and $y_2 = 2.787$ by manipulating u_1 , and u_2 . This step change test is carried out in the presence of disturbance on the inlet concentration C_{b1} , and physical constraints on the manipulated variables between 0 and 10. For this situation the concentration response suffers from excessive overshoot and slow disturbance rejection as reported by Brengel and Seider.

Tuning of the NLMPC parameters is necessary to reduce the overshoot, reject the disturbance, and maintain proper speed of the response. In order to impose these desired properties with CONSOLE, they are translated into transient upper and lower bounds on the response of the process outputs y_1 and y_2 . For y_2 tight upper bounds of 2.86 and 2.8 for the intervals 0-10 and 10-30 respectively are employed to prevent overshoot. Lower bounds of 2.6 and 2.73 for the intervals 5-10 and 10-30 respectively are also imposed to avoid sluggishness and ensure convergence of the response to its final steady state. For y_1 a constant upper bound of 101 and a lower bound of 99 from time

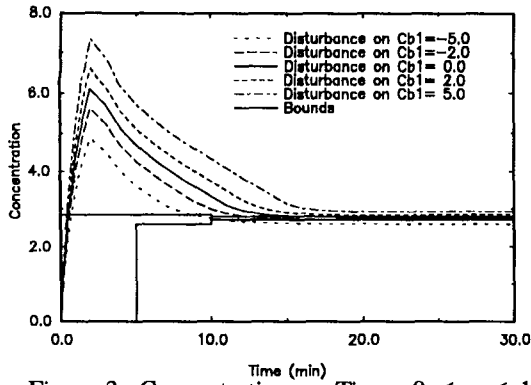


Figure 3: Concentration vs. Time. $0 \leq u \leq 10$, $M = 1$, $P = 3$, $\sigma = 0$, $D = \text{diag}[0,0]$, $\Gamma = \text{diag}[1,1]$.

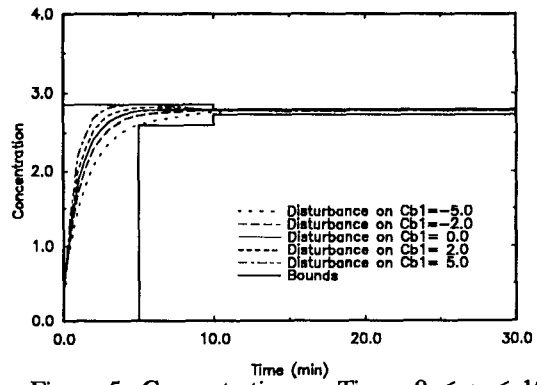


Figure 5: Concentration vs. Time. $0 \leq u \leq 10$, $M = 1$, $P = 3$, $\sigma = 0.087$, $D = \text{diag}[15.4,0]$, $\Gamma = \text{diag}[1,108.6]$.

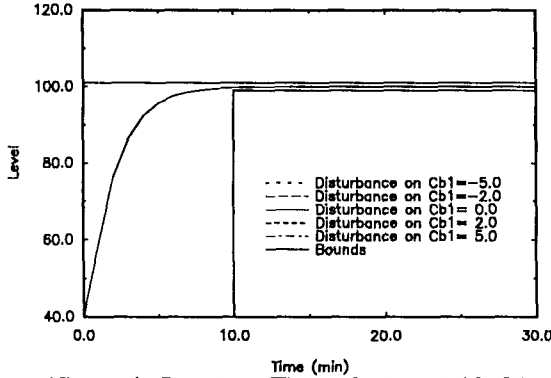


Figure 4: Level vs. Time. $0 \leq u \leq 10$, $M = 1$, $P = 3$, $\sigma = 0$, $D = \text{diag}[0,0]$, $\Gamma = \text{diag}[1,1]$.

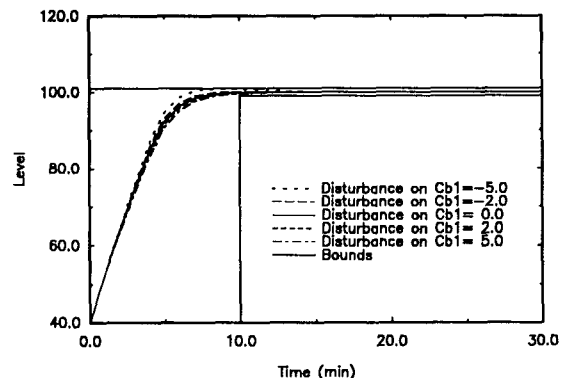


Figure 6: Level vs. Time. $0 \leq u \leq 10$, $M = 1$, $P = 3$, $\sigma = 0.087$, $D = \text{diag}[15.4,0]$, $\Gamma = \text{diag}[1,108.6]$.

10 to 30 were used to avoid performance degradation due to penalty weight variation. The performance bounds are represented by the solid lines in Figures 3 to 6.

Grid search along with CONSOLE is used to determine the optimal values of M , P , σ , D , and the weight on the second output (Γ_2) that simultaneously force the responses of the setpoint change under three different values of disturbances to fulfill the performance constraints. In particular step disturbances of magnitude 5.0, 0, and -5.0 on C_{b1} were used. The results of this search, which uses a sampling time of 1min, are summarized in Table 1. At each fixed value of M and P , The parameters σ , D , and Γ_2 in Table 1 are the final values found by running CONSOLE starting with initial values of 0.0, $\text{diag}[0,0]$, and 1.0 respectively. The table indicates whether the three responses at the obtained parameter values satisfy the bounds or not. In case they are not, the corresponding performance is then the best achievable one for that case. The table also shows the total required NLMPC simulations which have been used for each grid point.

For small values of P the response is so aggressive that CONSOLE was not able to move from the initial values of the tuning parameters. For large values of M and P the response of y_1 is slower and further decrease of the overshoot was not possible because it forces the response to violate the lower bounds. Only at $M = 1$ and $P = 3$ optimal tuning parameters that successfully satisfy the desired

objectives were found. Figures 3 and 4 show the setpoint responses of y_1 and y_2 at $M = 1$, $P = 3$, and initial values of the tuning parameters. While Figures 5 and 6 show the responses for the same values of M and P but at the final values of the tuning parameters. The figures also show the responses at additional disturbances values of +2.0 and -2.0. Since this system has slow dynamics it takes longer time to settle down to its final steady state value which would be clear if longer simulation time was used in the figures. Finally one should note that the NLMPC simulations required by CONSOLE would have been reduced by about one third, if the case of zero disturbance on C_{b1} was not used in CONSOLE.

Table 1.

M=1					
P	Γ_2	σ	D [diag]	B	N
1	1.0	0.0	[0,0]	No	12
2	1.9	4.0e-9	[0,0]	No	12
3	108.6	0.087	[15.46,0]	Yes	147
4	18.44	0.097	[0,0]	No	129
5	168.6	0.109	[1.8,5.3]	No	363
6	18.83	0.134	[1.85,13.2]	No	255
7	9.88	0.157	[21.6,13.3]	No	328

B=Bounds satisfied, N=NLMPC simulations

5. Discussion

This paper considered the question of tuning a nonlinear Model Predictive Control algorithm. A differential equation model is assumed for the process and an Extended Kalman Filter is incorporated to allow stable operation around open-loop unstable steady state points and better disturbance rejection. The use of the software package CONSOLE proved very effective in obtaining solutions to an off-line optimization problem used to tune the NLMPC parameters. One can expect the technique to work even more efficiently when applied to problems in which simulations of MPC algorithms are less computationally demanding, as, for example, the case of discrete nonlinear models, link neural networks, or linear models. Although in the case of linear models theoretical design techniques are available, the time-domain orientation of this approach may be preferable to a designer.

The integer nature of the control (M) and prediction (P) horizons create difficulties for the off-line optimization and require a grid search. The situation can be improved by treating P as a real variable by a simple modification of the on-line objective function. The use of differential equation model allows prediction at non-integer P . For cases where the model prediction is not computationally demanding, an alternative is to select a large fixed P . The formulation for non-integer P is given in Ali and Zafriou [1], where also certain stability questions are discussed in the context of satisfying the off-line time domain specification.

Acknowledgements

Support for this project is provided by the National Science Foundation (PYI grant CTS-9057292), the Institute for Systems Research, and grants from Shell and Texaco. The authors are grateful to Dr. Jian Zhou and Prof. André Tits for making the CONSOLE software available, and for their help in using it.

References

- [1] Ali, E., and Zafriou, E., "Optimization-Based Tuning of Nonlinear Model Predictive Control with State Estimation", submitted to the *J. of Process Control*, 1993.
- [2] Bequette, B. W. "Nonlinear Control of Chemical Processes- A Review", *Ind. & Eng. Chem. Res.*, **30**, 1391-1413, 1991.
- [3] Sistu, P. B., and Bequette, B. W. "Nonlinear Predictive Control of Uncertain Processes: Application to a CSTR", *AICHE J.*, **37**, 1711-1723, 1991.
- [4] Brengel, D. D., and Seider, W. D. "Multistep Nonlinear Predictive Controller", *Ind. & Eng. Chem. Res.*, **28**, 1812-1822, 1989.
- [5] Eaton, J. W., and Rawlings, J. B. "Feedback Control of Chemical Processes Using On-line Optimization Techniques", *Comp. and Chem. Eng.*, **14**, 469-479, 1990.
- [6] Garcia, C. E. "Quadratic Dynamic Matrix Control of Nonlinear Processes: An Application to a Batch Reaction Process", *AICHE ann. mtg.*, San Francisco, CA, 1984.
- [7] Gattu, G., and Zafriou, E. "Nonlinear Quadratic Dynamic Matrix Control with State Estimation", *Ind. & Eng. Chem. Res.*, **31**, 1096-1104, 1992.
- [8] Gattu, G., and Zafriou, E. "On the Stability of Nonlinear Quadratic Dynamic Matrix Control", In *IFAC/DYCORD+ '92 Preprints*, College Park, MD, USA, 303-308, 1992.
- [9] Gill, P. E., Murray, W., Saunders, M. A., and Wright, M. H., "User's Guide for NPSOL (version 4.0): A Fortran Program for Nonlinear Programming", Technical Report SOL 86-2, Department of Operations Research, Stanford University, 1986.
- [10] Jang, S. S., Joseph, B., and Mukai, H. "Control of Constrained Multivariable Nonlinear Process Using Two-Phase Approach", *Ind. & Eng. Chem. Res.*, **26**, 2106-2114, 1987.
- [11] Lewis, F. L., "Optimal Estimation with Introduction to Stochastic Control Theory", Wiley and Sons, New York, 1986, 260.
- [12] Li, W. C., and Biegler, L. T. "Newton-Type Controllers for Constrained Nonlinear Processes with Uncertainty", *Ind. & Eng. Chem. Res.*, **29**, 1647-1657, 1990.
- [13] Li, W. C., Biegler, L. T., Economou, C. G., and Morari, M., "A Constrained Pseudo-Newton Control Strategy for Nonlinear Systems", *Comp. and Chem. Eng.*, **14**, 451-468, 1990.
- [14] Nye, W. T., and Tits, A. L., "An Application-Oriented, Optimization-based Methodology for Interactive Design of Engineering Systems", *Int. J. Cont.*, **43**, 1693-1721, 1986.
- [15] Petzold, L. R., "A Description of DASSL: A Differential-Algebraic System Solver", in *Scientific Computing*, R. S. Stepleman, Ed., North-Holland 1983.
- [16] Ricker, N. L. "Model Predictive Control with State Estimation", *Ind. & Eng. Chem. Res.*, **29**, 374-382, 1990.
- [17] Hamilton, J. C., Seborg, D. E., and Fisher, D. G. "An Experimental Evaluation of Kalman Filtering", *AICHE J.*, **19**, 901-908, 1973.
- [18] Tits, A. L., Fan, M. K., Zhou, J., Wang, L., and Koninckx, J., "Console User's Manual (version 1.1)", System Research Center, University of Maryland, Technical Report SRC-TR 87-212r2, College Park, MD, 1990.
- [19] Wright, G. T., Breedijk, T. B., and Edgar, T. F., "On-line Parameter Estimation and Adaptation in Nonlinear Model-Based Control", American Control Conference, Boston, MA, 2782-2787, 1991.
- [20] Zafriou, E., and Marchal, A. L. "Stability of SISO Quadratic Dynamic Matrix Control with Hard Output Constraints", *AICHE J.*, **37**, 1550-1560, 1991.
- [21] Zhou, J., "Fast, Globally Convergent Optimization Algorithms, with Application to Engineering System Design", *Ph.D. Dissertation*, Electrical Engineering Department, University of Maryland, College Park, MD, USA, 1992.